Calhoun: The NPS Institutional Archive

DSpace Repository

Information Technology and Communication Services (ITACS) Computer Facility Newsletter, 1968-1999

1981-02-18

Computer Center News Letter / February 18, 1981

Monterey, California, Naval Postgraduate School

http://hdl.handle.net/10945/57187

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun

http://www.nps.edu/library

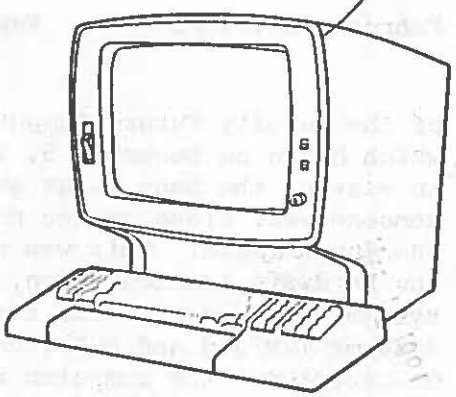
Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

469

D-5

Computer Center News Letter



NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

February 18, 1981
Volume 13, No. 2

TABLE OF CONTENTS

***

IBM 3033 - THE FIRST MONTH

As this is written, we have just completed the first month of operation with the IBM 3033AP system. It was put into general service on January 5, 1981, the beginning of the winter quarter. This was the culmination of efforts that began formally some four years ago with the formation

of the Faculty Future Computer Planning Committee. The system installation, which began on December 5, 1980, went according to plan, surprisingly so in view of the many snags and potential "disasters" experienced. Only someone very close to the problem would appreciate the magnitude of the task undertaken. This was not just a case of substituting processors in the hardware configuration, but involved a complete reorganization of the system configuration and the computer room, a change to two complex operating systems (VM/370 and MVS), and the introduction of new software products. On December 5 the computer room was completely cleared, right down to the underfloor area. The new system was literally built from the ground up. Talk about a point of no return! The accomplishment over the Christmas vacation was an outstanding combined effort, with superlative performance by the Center's staff, IBM engineers and Public Works personnel. Looking back now, some of the activities that spring to mind are

(1) fortnightly trips to Washington, D.C. at periods during the last year, and many late nights spent in the ADP Selection Office;

(2) dedication of the systems staff under Dave Norman's leadership, commuting to Palo Alto to generate and test VM/370 and MVS before installation at NPS;

(3) the outstanding efforts of User Services under Roger Hilleary in generating initial documentation and teaching classes while learning the system themselves;

(4) the anxious period from August to November 1980, during which we had a contract but no funds until Congress passed an FY81 continuing resolution;

(5) the night of December 5, when most of the DMDC staff came over to help us clear the computer room;

(6) Thursday, December 18, when the three temporary transformers which had been installed weeks before under the cypress trees were put under load and promptly collapsed. However, Public Works replaced them and had us running again within 24 hours;

(7) the Neptune truck bringing system components from New York being delayed by the California Highway Patrol;

(8) operators stringing what seemed like miles of coaxial cable in Ingersoll Hall;

(9) balance of the funds for the installation contract not being made available until two hours before the proposals expired.

Despite the substantial progress, a great deal of work remains. There is more hardware to install--an additional four megabytes of processor storage (February), graphics devices (June) and an additional eight spindles of IBM 3350 disk storage (July). One significant systems software task

remains--replacement of JES2 with JES3, which will allow improved two-way communication between the timesharing and batch-processing systems. In addition to restoring all the old faithful software products, we have introduced a lot of new facilities, including several packages from the University of Waterloo, viz. CMS WATFIV, BASIC, PASCAL and SCRIPT. (See Newsletter items below for details.) New IBM-supplied products include IPF (Interactive Productivity Facility), VS APL and VS COBOL.

The contract for the permanent site preparation was awarded on Friday, February 6, to Daniels & House Construction, Monterey. First priority will be the laying of cables for the terminals to be relocated in the academic buildings. There will also be a great deal of work at the west end of Ingersoll to complete the installation of the uninterruptible power system. Scheduled completion date for the project is September 9, 1981. We will make every effort to minimize the inconvenience to users.

MVS USERS CAN NOW USE MASS STORAGE

The Mass Storage System (MSS) is available to MVS batch users who wish to join with the Computer Center staff in developing this resource. Presently, any use of the MSS is on an "at your own risk" basis. Users who wish to assist us by providing a "real world" job load for MSS must recognize that this is an extremely complex hardware/software subsystem with which the Computer Center has only limited operational and systems programming experience. Presently, no data recovery or backup procedures are in place and any data loaded to MSS should be easily recoverable by the user.

MSS Access

MSS volumes are defined in "groups." Presently there are three general availability groups, each containing two volumes. Volume groups eliminate the need for specific volume serial numbers and implement other MSS management parameters. Additional volumes may be added to MSS volume groups as needed. Dataset space and retention dates are automatically assigned but may be overridden.

Each volume group has a unique name. Group names for general availability groups are in the format

```
PUBnl    where n = primary space allocation
                l = retention period
```

The following volume groups are presently available:

```
PUB4A    4 cyls     90 days
PUB4B    "  "      180 days
PUB4C    "  "      365 days
```

MSS dataset names must all begin with the high-level qualifier 'MSS' followed by your standard DSNAME.

Example:

```
        S
        F
MSS.Cnnnn.anything
        N
        X

        S  Student
        F  Faculty
        C  Computer Center
        N  Administration
        X  External
     nnnn  User Number
```

All MSS datasets must be cataloged.

To allocate a MSS dataset, use the following JCL:

```
//ddname  DD DISP=(NEW,CATLG),UNIT=3330V,
//           MSVGP=name,DSN=MSS.xnnnn.anything,
//           DCB=(dcb parameters)
```

To allocate a partitioned dataset, you must also include a space parameter that specifies directory block allocation:

```
SPACE=(CYL,(4,4,5))
```

If you need more space than the default, allocate in increments of four cylinders to ensure efficient use of the MSS. NEVER ALLOCATE IN TRACKS.

To access a previously allocated and cataloged MSS dataset, specify

```
                  SHR
//ddname  DD  DISP=OLD,DSN=MSS.xnnnn.anything
```

Files which do not follow the conventions specified herein (such as files without the high-level qualifier 'MSS' or files not allocated in four-cylinder increments) will be deleted by the Computer Center.

Users with a need for multiple short files which don't use enough space individually to justify a four-cylinder allocation should consider a single four-cylinder partitioned dataset instead. Please see Dave Norman or Kris Butler if you have questions about how to do this.

APL NOTES

VS APL, the new version of APL, is available on CMS. A suggested reference is the IBM booklet *VS APL Reference Summary*, SX26-3712-3. This booklet can be ordered from IBM for $1.45. For more information about public workspaces, see the notices posted on the door of In-102A.

APL users should look for the 3278 terminals which are equipped with the APL character set on the keyboard. The APL environment is invoked by typing

```
apl
```

The user should then change to the APL character set by holding down 'ALT' and pressing the large backspace key. The letters 'APL' will appear at the bottom of the screen.

ASCII Terminals

At this time, an ASCII terminal, such as a Miniterm, cannot be used to communicate with VS APL. Such a capability will require a modification of the operating system. See the bulletin board in the Consulting Office for any changes in this situation.

APL Hardcopy

Four IBM 2741 terminals will soon be installed next to the keypunches in the hallway of Ingersoll Hall. These terminals are hardwired into the IBM 3033AP specifically for use by APL programmers to obtain hardcopy printout of their programs and results.

SUBROUTINE LIBRARIES AVAILABLE UNDER VM AND MVS

In addition to the major statistical packages available to Computer Center users at NPS, there are three libraries which contain mathematical and statistical routines which may be called from Fortran programs. Two of these are the IMSL libraries (one each for double precision routines and single precision routines). Other useful routines not supplied by IMSL, Inc., are known at the Center as "non-IMSL routines" and are stored in the NONIMSL Library.

During the first three weeks of the new system these libraries were available only to batch (MVS) users. Now they are available to VM/CMS users as well.

To access any of these libraries under VM/CMS, the user must issue an appropriate 'global txtlib' command at the terminal prior to executing a Fortran job. All TXTLIBs which a program may access must be included in the 'global txtlib' statement. The names of the TXTLIBS containing the above-mentioned libraries are

IMSLSP	IMSL Single Precision Library
IMSLDP	IMSL Double Precision Library
NONIMSL	Selected Members from the NONIMSL Batch Library

A typical 'global txtlib' command for a Fortran user is as follows. Remember that TXTLIBs normally needed to execute a Fortran program must also be included in this statement:

```
global txtlib fortmod2 mod2eeh imslsp nonimsl
```

This command would allow the user to include calls to any IMSL single precision routines and to selected NONIMSL routines in his Fortran program.

A list of the NONIMSL routines that currently exist in the NONIMSL TXTLIB is given below. If you as a NONIMSL user find that you need a routine not included in this list, please notify Jane Foust in In-103 (ext. 2696). Keep in mind that routines calling the Versatec plotter cannot be executed from VM/CMS.

AND	FOURT	HISTG	OR	QSF	SECTN
BESJ	GBYTE	HISTGS	PLOTP	QZ	SORTHO
BOXPLX	GBYTES	HISTO	PLOTT	RANDU	SQUISH
CRVFIT	GELB	INTRPL	PLOT8	RHARM	UNPACK
DHARM	GELG	JACK	PXSORT	RKLDEQ	UTPLOT
DIFENT	HARM	LIST	QATR	RTMI	UTPLTT
DRHARM	HISTF	LLRANDOM*	QG4	RINI	UTPLT8
EXPLT	HISTFS	NORMPL	QG8		

*A new version.

PLEASE RETURN OLD USER'S MANUALS

All old User's Manuals from the days of the IBM 360/67 are now obsolete. These are usually contained in blue looseleaf binders with gold lettering. We are asking you to turn them in to the User Registration Office, In-147. We hope to issue a current manual sometime in the future.

ASCII MINITERMS

The following are some tips for the users of the portable CDI Miniterm:

Tabbing

Tabbing is a convenient method for indenting lines when you create a file. Filetypes have their own default tab settings. For example, Fortran filetypes have tabs automatically set at columns 1 7 10 15 20 25 . . . Tabs can be set with the XEDIT command SET TABS X X X X . . . where the X's represent the desired tab columns. To check where the tabs are set, use the XEDIT command Q TABS. These two commands can be given only in the XEDIT environment.

To tab on an ASCII terminal, hold down the CTRL key and then press the I key. If you want to move to the next tab setting, continue to hold down CTRL and press the I key again. The CTRL key is like a shift key. It must be down when you press I. When you tab, the printing mechanism will not move. But when you display the line you have typed in, the tabbing will be shown.

Line Length

While the terminal is capable of printing an 80-character line, the default line length is 72. This shorter line length causes 80 column lines, such as in HELP files, to be printed on two lines. To get the full 80-character line, use the command

```
cp terminal lines 80
```

For convenience, put this command in your PROFILE EXEC.

Serial Line Numbers

Some find the serial line numbers of Fortran programs annoying when printed out. Whenever you XEDIT a file with a Fortran filetype, serial numbers are automatically created unless you use this XEDIT command:

```
set serial off
```

For convenience, this command can be put in your PROFILE XEDIT. The command will then be executed every time you enter the XEDIT environment.

STATISTICAL PACKAGES ON MVS

The following three commercial statistical packages are available on the batch system:

SAS - Statistical Analysis System (version 79.2B)
SPSS - Statistical Package for the Social Sciences (version 8.1H)
BMDP - The Biomedical Statistical Package (version 79)

Job control language remains the same as before.

Both SPSS and SAS have sorting procedures which are automatically linked to the commercial sort routine, SYNCSORT. To use SPSS's SORT CASES, see the Computer Center Newsletter of 19 November 1980. To use SAS's PROC SORT, see the *SAS User's Guide* in the Consulting Office, In-146.

IODE IS NOW ON CMS

The Interactive Ordinary Differential Equation Solver (IODE) is now available for use on CMS. This package solves almost any initial value problem in differential equations by asking you questions, then providing the solutions. You can obtain a printed copy of the writeup by issuing:

```
iode d
```

A copy of the writeup can be viewed on any 3278 terminal by issuing:

```
iode h
```

You can initiate the solution of a problem by issuing:

```
iode
```

Solutions can be obtained in the form of tabular numerical output and/or as printer-style plots. Either form can be displayed at your terminal and/or on the system printer.

XEDIT NOTES

The purpose of this article is to share with VM/CMS users some of the features of the XEDIT editor which the Center's staff have found useful.

Use the "Set Nulls On" Option

When using the "insert" key (â) to insert text into a line, it is usually necessary to first press the PA2 key to clear the end of the line of blanks. By issuing the command

```
set nulls on
```

while in XEDIT mode, you can use the insert key conveniently without having to clear each line with the PA2 key. This command may be put into your PROFILE XEDIT.

A word of caution: When inserting at the end of a line, it is necessary to enter any desired blanks immediately following the last character on the line.

Not Enough Space on Your Disk to "File"?

Sometimes when your disk space is almost full, or when you XEDIT a very large file, there will not be enough space left on your disk to file after editing. In this case you must make room on your disk for the file, or lose the efforts of the editing session. Because the XEDIT editor allows you to issue CMS commands without leaving the XEDIT environment, you can usually correct the situation and preserve file changes as well. There are several options available. One suggestion:

To enter the CMS environment from XEDIT, type 'cms' (enter); then type any CMS commands; return to XEDIT by typing 'return' (enter).

To make room on a disk while in CMS, either

1) erase unnecessary files from the disk using the 'erase' command, or

2) erase the file being XEDITed, and refile the changed file.

The latter option is risky; if the system should "crash" before the new, XEDITed file is refiled on the disk, the file may be lost unless you have a backup.

Use the "Selective Change" Option

Two program function keys, PF5 and PF6, may be used alternately to "selectively change" a pre-specified string in the text of a file being XEDITed. For example, you wish to change the string 'bitter' to 'butter' in some places within the text, but not everyplace; and you wish to view each occurrence of the string before deciding if the change is to be made. Do this using the PF5 and PF6 keys in the following manner:

1) Type the change command on the command line:

```
===> change /bitter/butter/  (do not press ENTER!)
```

2) Press the PF5 key to locate the character string. The line will be highlighted and the column pointer will be positioned under the string.

3) Press the PF6 key only if the change is to be made.

Continue steps 2 and 3 until all desired strings have been changed or viewed.

One important note on the use of the PF keys for the selective change function is that only the PF keys at the top of the keyboard on the IBM 3278 terminal (those which must be used in conjunction with the ALT key) appear to correctly execute this feature. The PF keys at the right of the keyboard do not give the desired result.

SAVE PAPER WHEN PRINTING A VM/CMS FILE

When a VM/CMS file is sent to the printer (using the PRINT command), typically four pages are used in merely printing the identifying header and the trailer. If you have more than one file to print, a significant amount of paper can be wasted. However, you can eliminate the intermediate identification pages simply by using the following technique:

Before printing the first file, issue the CP command

```
spool printer cont
```

This will set the virtual printer in a continuous print mode. Next issue the print command for each of the files to be printed. At completion of printing, issue the following command to return the virtual printer to the noncontinuous printing mode (the default mode), if desired:

```
spool printer close nocont
```

Optionally, you may close the virtual printer, but leave the continuous print mode in effect for the terminal session by issuing:

```
spool printer close
```

Note also that "printer" may be abbreviated to "prt". Below is an example of a typical session when several files are to be printed on the printer:

```
spool prt cont
      (print file)
      (print file)
      (print file)
spool prt close
      (...continue terminal operations such as editing, etc.)
      (print file)
      (print file)
      (...end of printing for the session)
spool prt close nocont
```

For more information on the "spool" function, type the following command at a 3278 terminal:

```
help cp spool
```

UNPACK FUNCTION CAN BE USED

Users who had packed files on the CP/CMS system can now unpack them on VM/CMS. The new unpack module performs in exactly the same manner as the old:

```
unpack <filename>
```

The filetype must be "packed." The resulting file will have filetype "unpacked."

USING THE LOADER ON MVS

Whenever possible, we urge MVS users to use cataloged procedures involving the loader, rather than the linkage editor, e.g., Fortran users should use FORTXCG rather than FORTXCLG. (Incidentally, FORTCLG is now an alias of FORTXCG.)

The loader is a service program that combines linkage editing and execution in one step. Like the linkage editor, the loader accepts object modules passed to it from the compiler, resolves external references, and searches subroutine libraries.

The loader differs from the linkage editor in not producing a load module. Instead it processes all object modules in storage. Enough storage must be allowed for both the loader and the program. When processing is complete, the loader passes control to the program for execution. This process is usually called "load and go."

The advantage of the loader is its speed; it is about twice as fast as the linkage editor. Time is also saved by the reduced overhead of a single job step. The disadvantage is that a load module cannot be saved for later execution.

Also, at NPS, programs that involve Versatec plotting must use the linkage editor.

For more information on cataloged procedures that invoke the loader, see the NPS publication "Questions and Answers for MVS (Batch) Users."

WATFIV IS AVAILABLE

WATFIV, the one-step compile-and-execute Fortran processor from the University of Waterloo, is now available on both CMS and MVS at the School. Special features of WATFIV include:

- Statements to support structured programming, such as IF-THEN-ELSE-ENDIF, WHILE and QUIT
- Format-free I/O
- Character variables
- Excellent compile and run-time diagnostics

In addition, CMS WATFIV has an interactive debugging facility.

MVS WATFIV

To use MVS WATFIV, submit a deck of the following form:

```
//  (standard JOB card)
//  EXEC WATFIV
//SYSIN  DD  *
$JOB

   WATFIV source statements

$ENTRY  (not optional)

   data (optional)

$$
/*
```

CMS WATFIV

To use CMS WATFIV, create one or more files: file1, file2, ...filen, which, when concatenated, will be your WATFIV program including $JOB and $ENTRY cards and data. Each file must have filetype either WATFIV or FORTRAN. Then issue the command

```
watfiv file1 file2 ... filen (options
```

where the options are of course optional. Any of the filenames can be replaced by a *, in which case the input will be read from the terminal.

Documentation

A copy of the WATFIV User's Guide is available in the Consulting Office, In-146, for reference. It describes the language accepted by WATFIV, the parameters which may be included on the $JOB card, and the options pertaining to CMS WATFIV.

You may obtain a printed copy of that part of the manual concerned with CMS WATFIV by issuing the following CMS command:

```
print watfiv listing y (up
```

SCRIPT HAS BEEN INSTALLED

The document composition program SCRIPT is available on CMS. The version installed originates from the University of Waterloo.

SCRIPT accepts as input the text you want to appear in your final document, interspersed with SCRIPT control words. As output, it produces the text according to the specifications indicated by the control words. SCRIPT can handle anything from a simple letter to a complex document such as a book or thesis, with footnotes, tables of contents, etc.

SYSPUB and SYSPAPER

To format many small documents, the full power of SCRIPT is not necessary. SYSPAPER and SYSPUB (each consisting of a limited number of SCRIPT macros) will suffice. SYSPAPER and SYSPUB are less flexible than the full SCRIPT language, but considerably easier to learn.

Using SCRIPT

To get terminal or disk output from SCRIPT, simply give the command

```
script <filename> (options
```

For printed output, give the command

```
scriptp <filename> (options
```

For an explanation of the options available with SCRIPT, we suggest you first issue the command

```
script ?
```

To get a list of all SCRIPT commands, issue the command

```
scriptp scriptcd (pr
```

It may take a day for your output to be printed, since the TN print chain must be mounted on the IBM 1403 printer. If you are willing to settle for an uppercase-only printout of the commands, and have sufficient disk space (1250 records of 74 bytes each), you can give the following commands to get a copy of your document quickly:

```
script scriptcd (disk
print scriptcd listing (up
```

PASCAL IS AVAILABLE ON CMS

Waterloo PASCAL, a one-step compile-and-execute PASCAL processor written at the University of Waterloo, is available on CMS.

The language accepted by Waterloo PASCAL is, with few exceptions, that defined by Jensen and Wirth's *Pascal User Manual and Report*. PASCAL is an ALGOL-like language with plenty of statements to support structured programming (e.g. IF-THEN-ELSE, CASE, WHILE) and a rich selection of data structures including the ability to invent your own. Functions and procedures may be called recursively. Data structures may be created dynamically during the execution of your program by using the POINTER data type and the standard function NEW.

Using PASCAL

To use PASCAL, create a file which is a PASCAL program. It must have filetype PASCAL. Then issue the command

pw <filename>

Documentation

A copy of the *Waterloo PASCAL User's Guide and Language Description* is available for reference in the Consulting Office, In-146. To obtain your own printed copy, issue the commands

print wpasuser listing y (up

This document assumes familiarity with the PASCAL language; it discusses the special features of the Waterloo implementation of PASCAL.

WATERLOO BASIC IS UP

Waterloo BASIC is now available on CMS. For information on how to use it, consult the manual in the Consulting Office, In-146, or give the command

pr basintro listing y (up

BASIC has its own HELP facility. To use it, type

wbasic

You will get a READY message back from BASIC (not from CMS). Then type

help

To exit BASIC, type

bye

The BASIC Environment

BASIC, like APL, was conceived initially as an interactive language; Waterloo BASIC has no batch version. The BASIC environment includes, in addition to a programming language, its own editor and some commands to handle functions normally accomplished through CMS.

You can use just the programming language part of BASIC by issuing commands of the form

wbasic <filename> (options

where the file is an EXEC or a BASIC program which has been created using any available editor.

You can enter the BASIC environment for an indefinite period of time by typing

wbasic

The BASIC Language

Special features of the Waterloo BASIC language include:

- Flexible processing of characters
- Ability to clear the screen under program control
- Ability to issue CMS subset commands under program control
- Interactive debugging: If an error is encountered in processing, the user is given control and may edit the offending statement, print the value of variables, transfer control to another part of the program, etc.

USING SYNCSORT UNDER MVS

SYNCSORT, the sort utility package marketed by Whitlow Computer Systems, Inc., of New Jersey, is now available on MVS. SYNCSORT is reported to be 20 to 80 per cent more efficient (depending on the application) than the IBM SORT-MERGE utility.

All user programs requiring a sort must have the following JCL card included in the GO step pointing to the sort library:

```
//GO.STEPLIB DD DSN=SYS1.SYNCSORT,UNIT=3350,
//  VOL=SER=MVS003,DISP=SHR
```

This card replaces the following card (if present), used with procedures on the IBM 360/MVT system:

```
//SORTLIB  DD  DSN=SYS1.SORTLIB,DISP=SHR
```

Most users with datasets requiring sorting will probably be able to use the cataloged procedure SORTD. The following is an example of the use of SORTD:

```
// (standard JOB card)
// EXEC SORTD
//SORTIN DD DSN=xxxxxxx,UNIT=xxxx,VOL=SER=xxxxx,
//  DISP=SHR,SPACE=(CYL,(X,X)),
//  DCB=(LRECL=xx,BLKSIZE=xxxx,RECFM=xx)
//SORTOUT DD DSN=xxxxxxx,UNIT=xxxx,VOL=SER=xxxx,
//  DISP=SHR,SPACE=(CYL,(X,X)),
//  DCB=(LRECL=xx,BLKSIZE=xxxx,RECFM=xx)
//SYSIN DD *,DCB=BLKSIZE=80,
//  SORT FIELDS=(xx,x,xx,x)
```

where: x's represent user-supplied variable information,
SORTIN represents the input data set,
SORTOUT represents the output data set,
SYSIN supplies sort control cards.

References

A copy of the User's Manual published by Whitlow, Inc. can be consulted in Neil Harvey's office, In-108. Users wishing to have their own copies can obtain them from Whitlow Computer Systems, Inc., attn: OS Technical Services, 560 Sylvan Ave., Englewood Cliffs, NJ 07632. Prices: Programmer's Guide, $15 with binder, $9 without binder; reference card, 40 cents, and an Application Guide, "Exploiting SYNCSORT," no charge.

OS/VS COBOL

COBOL OS/VS 2.3 is installed and ready for use on MVS. VS COBOL offers support for the 1974 ANS COBOL language standards. Two of the most common conversion problems encountered at NPS are due to (1) an expanded reserve word list in the new COBOL, and (2) the use of the double quote (") as the default for specifying literals. For a quick fix on the double quote problem, users may code an additional parameter on their EXEC card directing the compiler to accept the single quote. For example, when doing a compile, code

```
// EXEC COBUC,PARM.COB=(APOST)
```

Users are also cautioned that in OS VS COBOL, (1) the format of the COPY statement has been changed, and (2) that the IF statement functions differently for implied subject and relational operators.

A technical memorandum providing more information is available in In-146.

RECENT ADDITIONS TO THE COMPUTER CENTER LIBRARY

Books

Author	Title
Tucker, Allen B.	Text Processing - Algorithms, Languages and Applications
Yovits, Marshall C.	Advances in Computers, Vols. 17, 18 & 19
Demillo, R. et al	Foundations of Secure Computation
Heaps, H. S.	Information Retrieval - Computational & Theoretical
Hsiao, David K.	Computer Security
Organick, E. I. et al	Programming Language Structures
Conway, R. et al	Programming for Poets
Winkel, D. & Prosser, F.	The Art of Digital Design
Lukoff, Herman	From Dits to Bits - A History of Electronic Computers
Kennedy, W. J.	Statistical Computing
Holt, R. & Hume, J.	Programming Standard PASCAL
Brown, Gary D.	System 370 - Job Control Language
Hughes, C. E. et al	Advanced Programming Techniques
Olle, T. W.	The Codasyl Approach to Data Base Management
Davies, D. W.	Computer Networks and Their Protocols

Myers, Glenford J.	Advances in Computer Architecture
Cowell, Wayne	Portability of Numerical Software
Metropolis, N.	A History of Computing in the 20th Century
Goldstine, Herman H.	A History of Numerical Analysis, 16th-19th Century
Schneiderman, Ben	Software Psychology
NAVEDTRA	Introduction to Matter, Energy and Direct Current
NAVEDTRA	Introduction to Alternating Current and Transformers
Hofstadter, Douglas	Gödel, Escher, Bach: An Eternal Golden Braid

Reports

No.	Author	Title
1361	Wolcott, N.	Hershey's Occidental Type Fonts and Graphic Symbols
1362	Ware, W. H.	Security Controls for Computer Systems
1363	Ware, W. H.	Computer Security in Civil Government and Industry
1364	Smith, H. T. et al	An Office Communication System Design
1365	NTIS	Current Published Searches
1366	GAO	Millions in Savings Possible - Converting Programs
1367	Brown, W. S.	A Simple But Realistic Model of Floating-Point Computation
1368	Bell Labs	PIC - A Crude Graphics Language for Typesetting
1369	Bell Labs	A Bibliography on the Evaluation of Mathematical Functions

Proceedings

Organization	Title
SHARE, Inc.	Proceedings of SHARE 51, Vols. I & II - 1978
SHARE, Inc.	Proceedings of SHARE 53, Vols. I & II - 1979
SHARE, Inc.	Proceedings of SHARE 54, Vols. I & II - 1980
ACM	Proceedings of the 17th Annual Computer Research Conference, 1980
ACM	Proceedings of SIGGRAPH '80 Conference, 1980
ACM	Proceedings of Performance 80, 1980
ACM	Proceedings of the 3rd Symposium on Small Systems, 1980

The Newsletter appears semiquarterly and is written by members of the staff, W. R. Church Computer Center (Code 0141), Naval Postgraduate School, Monterey, California 93940. Requests for further information or suggestions for articles for the Newsletter may be addressed to the User Services Manager, Code 0141 (In-133), ext. 2752 (or ext. 2573 for messages).

The Center provides batch-processing service under OS/VS2 MVS with JES2 and time-sharing service under IBM VM/SP CMS. These services are based on an IBM 3033 Attached Processor system with 12Mbytes of storage.

Distribution

List 3, plus: 1-A5, 200-B2, 3-B3, 1-B13, 3-F3, 1-F4, 1-F5, 1-F6, 1-Code 49, 60-Student Mail Center (Lobby)